

Amendments to the Claims

Please replace the Claims as shown below:

1. (currently amended) A method of programmatically selecting system designs from a system design space, the method comprising:

specifying system designs as combinations of component designs from respective component design spaces;

applying component quality filters to the component design spaces to produce component ~~quality~~ Pareto sets of designs; and

forming a Cartesian product of the component ~~quality~~ Pareto sets to obtain a set of system designs.

2. (currently amended) The method of claim 1, further comprising applying component validity filters to respective component design spaces before applying the component quality filters, wherein the component ~~quality~~ Pareto sets of designs include only designs satisfying respective component validity filters.

3. (original) The method of claim 1, further comprising applying a system validity filter to the set of system designs to produce a validity filtered set of system designs.

4. (original) The method of claim 3, further comprising applying a system quality filter to the set of system designs.

5. (original) The method of claim 1, further comprising applying a system quality filter to the set of system designs.

6. (currently amended) A method of programmatically selecting system designs that are specified by combinations of component designs, the method comprising:

preparing component validity sets for each of the component designs by applying component validity filters to corresponding component design spaces, the component validity filters defined by corresponding component validity predicates; [[and]]

forming a set of system designs that is a Cartesian product of the component validity sets[[.]] ; and

applying a system quality filter to the Cartesian product of the component validity sets to produce a Pareto set.

7. (original) The method of claim 6, wherein the component designs are specified by component parameters, and the component validity filter for each component is independent of the component parameters of other components.

8. (original) The method of claim 6, further comprising applying a system validity filter to the Cartesian product of the component validity sets.

9. (currently amended) The method of claim 6, further comprising applying a ~~system quality filter to the Cartesian product of~~ component evaluation function to the component validity sets.

10. (currently amended) The method of claim 6, further comprising applying a system evaluation function and ~~[[a]]~~ the system quality filter to the Cartesian product of the component validity sets after applying a system validity filter.

11. (original) The method of claim 10, further comprising applying a component evaluation function and a component quality filter to the component validity sets.

12. (original) The method of claim 6, further comprising applying a component evaluation function and a component quality filter to at least one of the component validity sets before forming the set of system designs.

13. (original) The method of claim 12, further comprising:

- selecting a partial system design that includes component designs for at least one component;
- obtaining a lower bound for an evaluation metric for a system design, wherein the system design includes the partial system design; and
- comparing an evaluation metric of a system that includes the partial system design to the lower bound.

14. (currently amended) A method of selecting system designs that are specified by combinations of component designs, the method comprising:

preparing component validity sets for each of the component designs by applying component validity filters to corresponding component designs, the component validity filters defined by corresponding component validity predicates;

preparing component ~~quality~~ Pareto sets by applying corresponding component evaluation functions and component quality filters to the component validity sets; and

forming a set of system designs that is a Cartesian product of the component ~~quality~~ Pareto sets.

15. (currently amended) The method of claim 14, further comprising applying a system validity filter to the Cartesian product of the component ~~quality~~ Pareto sets.

16. (currently amended) The method of claim 14, further comprising applying a system evaluation function and a system quality filter to the Cartesian product of the component ~~quality~~ Pareto sets.

17. (original) The method of claim 16, wherein the component evaluation functions and the system evaluation function produce component evaluation metrics and system evaluation metrics, respectively, and the system evaluation metrics are obtained from the component evaluation metrics.

18. (original) A computer readable medium comprising computer executable instructions for performing the method of claim 1.

19. (original) A computer readable medium comprising computer executable instructions for performing the method of claim 6.

20. (original) A computer readable medium comprising computer executable instructions for performing the method of claim 14.

21. (currently amended) A method of programmatically selecting a system design from a set of system designs, comprising:

defining a system validity predicate that is a function of two or more terms;

defining partial validity predicates by expressing the system validity predicate in a canonical form;

applying partial validity filters that are defined by the partial validity predicates to the system designs to obtain partial validity sets; [[and]]

applying a quality filter to the system designs of the partial validity sets to produce respective partial Pareto sets; and

combining the designs from the partial validity Pareto sets to obtain sets of designs satisfying each of the two or more terms form a Pareto set.

22. (currently amended) The method of claim 21, ~~[[where]]~~ wherein each of the partial validity predicates is in product form.

23. (original) The method of claim 21, wherein the partial validity predicates are mutually exclusive.

24. (currently amended) A method of programmatically selecting a set of system designs, comprising:

selecting a system validity filter defined by a system validity predicate, the system validity predicate including one or more partial validity predicates that define partial validity filters;

applying the partial validity filters to the system designs;

forming partial validity sets that include system designs satisfying respective partial validity filters;

applying an evaluation function to the system designs of the partial validity sets, the evaluation function producing an evaluation metric for each system design;

applying a quality filter to the system designs of the partial validity sets, the quality filter comparing and selecting system designs based on the evaluation metrics and producing respective partial ~~quality~~ Pareto sets; and

combining the partial ~~quality~~ Pareto sets to form a first ~~quality~~ Pareto set.

25. (currently amended) The method of claim 24, further comprising applying the quality filter to the first ~~quality~~ Pareto set.

26. (original) The method of claim 24, wherein each of the partial validity predicates is in product form.

27. (original) The method of claim 26, wherein the system validity predicate is a product of the partial validity predicates.

28. (original) The method of claim 26, wherein the partial validity sets are combined to form two or more system validity sets.

29. (currently amended) A computer readable medium having computer ~~exeectuable~~ executable instructions for performing the method of claim 24.

30. (original) A computer readable medium having software for performing the method of claim 25.

31. (original) A method of programmatically selecting a design for a cache memory, comprising:

selecting components for the cache memory;

determining component Pareto sets for the components;

preparing a combined Pareto set from the component Pareto sets; and
selecting a cache memory design from the combined Pareto set.

32. (original) A method of selecting a design for a processor system, the processor system including a processor and a cache memory, the method comprising:
preparing a component Pareto set for the processor;
preparing a component Pareto set for a cache memory;
preparing a combined Pareto set from the component Pareto sets of the processor and the cache memory; and
selecting a processor system design from the combined Pareto set.

33. (currently amended) A method of programmatically generating a set of designs for a processor system, comprising:
dividing the processor system into at least a processor component and a memory component;
preparing component validity sets for the processor component and the memory component;
preparing component Pareto sets for the processor component and the memory component; and
forming a Cartesian product of the component ~~validity~~ Pareto sets to produce the set of designs for the [[a]] processor system validity set.

34. (currently amended) The method of claim 33, further comprising ~~expressing the system validity function in a logical canonical form~~ applying a system evaluation function and a system quality filter to the Cartesian product of the component Pareto sets.

35. (currently amended) A method of designing a processor system that includes a processor component and a memory component, comprising:

determining component validity sets for the processor component and the memory component;

dividing at least one of the component validity sets into subsets; [[and]]

~~generating sets of system designs by combining component designs from the component validity sets and the subsets.~~

applying component quality filters to the component designs from the component validity sets and the subsets to produce component Pareto sets of designs; and

forming a Cartesian product of the component Pareto sets to obtain a set of system designs.

36. (currently amended) A method of generating a set of partial validity predicates for a system design that includes component designs for at least a first component and a second component, the method comprising:

obtaining a system validity function defined by a system validity predicate; [[and]]

identifying coupled terms in the system validity predicate, the coupled terms including parameters of the components [[.]] ; and
producing a set of system designs that is a Cartesian product of component Pareto sets.

37. (currently amended) The method of claim 36, wherein the system design is a processor system design and the components include a processor component and a memory component.

38. (original) The method of claim 37, further comprising expanding the coupled terms to obtain singleton terms containing parameters of only the processor component and singleton terms containing parameters of only the memory component.

39. (currently amended) The method of claim 36, further comprising expanding the coupled terms to obtain singleton terms containing parameters of only [[a]] the first component and singleton terms containing parameters of only [[a]] the second component.

40. (original) The method of claim 39, further comprising expressing the system validity predicate in canonical form.

41. (original) The method of claim 36, further comprising expressing the system validity predicate in canonical form.